

**IN THE CLAIMS:**

The following is a listing of the claims as currently pending in this application.

- 1           1. (Original) In a computer system having one or more processors and one or  
2 more peripheral devices connected to an Input/Output (I/O) bus, an I/O bridge coupled to  
3 the I/O bus and configured to interface between the one or more processors and the one  
4 or more peripheral devices, the I/O bridge comprising:  
5           a queue for buffering information received from the one or more processors; and  
6           a transaction engine operably coupled to the queue, the transaction engine config-  
7 ured to place information buffered at the queue onto the I/O bus for receipt by a targeted  
8 peripheral device,  
9           wherein the transaction engine:  
10               generates an attribute message that includes a tag field and a requester  
11               function number field,  
12               loads the tag field with a selected value,  
13               loads the requester function number field with a selected one of a plurality  
14               of values, and  
15               places the attribute message including the selected tag and requester func-  
16               tion number values onto the I/O bus for receipt by the targeted I/O device.
- 1           2. (Original) The I/O bridge of claim 1 wherein    the transaction engine logi-  
2 cally concatenates the tag field and the requester function number field of the attribute  
3 message to create a super tag value for use in tracking transactions placed on the I/O bus.
- 1           3. (Original) The I/O bridge of claim 2 wherein the super tag ranges from binary  
2 “00000000” to binary “11111111”.

1           4. (Original) The I/O bridge of claim 1 wherein the queue has a plurality of entries  
2 for buffering the information and each queue entry is associated with a corresponding tag  
3 value and a corresponding requester function number value.

1           5. (Original) The I/O bridge of claim 4 wherein, in response to a Split Completion  
2 Message containing a tag value and a requester function number value, the transaction  
3 engine uses the received tag and requester function number values to identify a corre-  
4 sponding queue entry and clears the identified entry.

1           6. (Original) The I/O bridge of claim 5 wherein the I/O bus operates in substantial  
2 compliance with the Peripheral Component Interface Extended (PCI-X) specification  
3 standard.

1           7. (Original) The I/O bridge of claim 4 wherein the transaction engine is further  
2 configured to place information received from a peripheral device along with a Split  
3 Completion transaction that specifies a tag value and a requester function number value  
4 into the queue entry associated with the specified tag and request function number values.

1           8. (Original) The I/O bridge of claim 7 wherein the information buffered at the  
2 queue comprises at least one of command, address and data, and the command may be  
3 read or write.

1           9. (Original) The I/O bridge of claim 1 wherein the queue includes a read buffer  
2 for buffering data that was received from a peripheral device and a write buffer for buff-  
3 ering information that is to be provided to a targeted peripheral device.

1           10. (Original) The I/O bridge of claim 1 wherein the I/O bus operates in substan-  
2 tial compliance with the Peripheral Component Interface Extended (PCI-X) specification  
3 standard.

1           11. (Original) A method for use in a computer system having one or more proces-  
2           sors, one or more memory subsystems, and one or more peripheral devices connected to  
3           an Input/Output (I/O) bus, the method comprising the steps of:

4                 providing at least one queue having a plurality of entries for buffering information  
5                 received from or to be sent to a targeted peripheral device;

6                 associating each queue entry with a selected tag value and with one of a plurality  
7                 of selected requester function number values;

8                 buffering information received from a processor or a memory subsystem in a se-  
9                 lected queue entry;

10                generating an attribute message that includes a tag field and a requester function  
11                number field;

12                loading the tag field of the attribute message with the tag value associated with  
13                the selected queue entry;

14                loading the requester function number field of the attribute message with the re-  
15                quester function number value associated with the selected queue entry; and

16                placing the attribute message including the tag and requester function number  
17                values onto the I/O bus for receipt by the targeted I/O device.

1           12. (Original) The method of claim 11 further comprising the steps of:

2                receiving a Split Completion transaction from a targeted peripheral device speci-  
3                fying a tag value and a requester function number value and including data;

4                utilizing the received tag and requester function number values as an index to  
5                identify a corresponding queue entry; and

6                buffering the data received from the targeted peripheral device at the identified  
7                queue entry.

1           13. (Original) The method of claim 11 further comprising the step of logically  
2           concatenating the tag field and the requester function number field of the attribute mes-  
3           sage to create a super tag value for use in tracking transactions placed on the I/O bus,  
4           wherein the super tag ranges from binary "00000000" to binary "11111111".

1           14. (Original) The method of claim 13 wherein the I/O bus operates in substantial  
2 compliance with the Peripheral Component Interface Extended (PCI-X) specification  
3 standard.

          15. (Original) The method of claim 14 wherein the information buffered at the  
queue comprises at least one of command, address and data, and the command may be  
read or write.